JAVA LAB PROGRAMS

1) Program to generate two threads, one for printing even numbers and the other to print odd numbers.

```
public class Q1 extends Thread {
	Thread th;
	int[] nums = new int[10];
	Q1(int[] nums, String name) {
		th = new Thread(this, name);
		this.nums = nums;
		th.start();
	}
	public void run()  {
		for(int i = 0; i < 10; i++) {
			System.out.println(th + ": " + nums[i]);
		}
	}
	public static void main(String args[]) {
		int even[] = new int[10];
		int odd[] = new int[10];
		int i = 0, j = 0;
		for(int k = 1; k <= 20; k++) {
			if(k % 2 == 0)
				even[i++] = k;
			else
				odd[j++] = k;
		}
		new Q1(even, "Even");
		new Q1(odd, "Odd");
	}
}
```

2) Program to create a Base class and Derived class having data and overridden function members. Demonstrate the use of Dynamic Method Dispatch.

```
public class Q2 {
	public static void main(String args[]) {
		A a = new A();
		B b = new B();
		C c = new C();
		A r;
		r = a;
		r.callMe();
		r = b;
		r.callMe();
		r = c;
		r.callMe();
	}
}
class A {
	void callMe() {
		System.out.println("Inside A's callMe method");
	}
}
class B extends A {
	void callMe() {
		System.out.println("Inside B's callMe method");
```

```
	}
}
class C extends A {
	void callMe() {
		System.out.println("Inside C's callMe method");
	}
}
```

3) Create a class customer for a bank. Class account should inherit the properties of the customer and having 'type of account' and 'interest rate' as members. Use constructors in base and derived class to initialize the data members. An account may be created with zero initial deposit. (Use default arguments, super keyword). Implement the methods deposit(), withdraw() and getBalance() to operate an account.

```
public class Q3 {
	public static void  main(String args[]) {
		Account a1 = new Account("AAA", 22, "Savings", 2.5);
		Account a2 = new Account("XYZ", 23, "Current", 2.2);
		a1.deposit(1000);
		a2.deposit(1500);
		a1.withdraw(200);
		a2.withdraw(600);
		System.out.println("Balance of " + a1.name + ": " + a1.getBalance());
		System.out.println("Balance of " + a2.name + ": " + a2.getBalance());
	}
}
class Customer {
	String name;
	int cId;
	Customer(String a, int b) {
		name = a;
		cId = b;
	}
}
class Account extends Customer {
	String type;
	double rate;
	long balance = 0;
	void deposit(long amount) {
		balance += amount;
	}
	void withdraw(long amount) {
		balance -= amount;
	}
	long getBalance() {
		return balance;
	}
	Account(String a, int b, String c,  double d) {
		super(a, b);
		type = c;
		rate = d;
	}
}
```

4) Define a Person class with three data members' age, name and gender. Derive a class called Employee from Person that adds a data member 'emp_code' to store employee code. Derive another class called Specialist from Employee. Add a method to each derived class to display the information about what it is. Write a program to generate an array of three ordinary employees and another

array of three specialists and display the information about them. Also display the information of the specialists by calling the method inherited from employee class.

```
package part_a;

import java.util.*;
class Person {
	int age;
	String name;
	String gender;

	void display_person() {
		System.out.println("Name = "+name);
		System.out.println("Age = "+age);
		System.out.println("gender = "+gender);
	}
}

class Employee extends Person
{
	int emp_code;

	void display_emp()
	{

		System.out.println("The Employee code of the Employee is "+emp_code);
	}
	Employee() {

	}
	Employee(String name1, int age1, String gender1, int code1)
	{
		Scanner read = new Scanner(System.in);

		name = name1;

		age = age1;

		gender = gender1;

		emp_code = code1;

	}
}

class Specialist extends Employee
{
	String specialist;


	void display_spec()
	{
		System.out.println("Specialist of  "+specialist);
	}

	Specialist(String n, int a, String g, String s) {
		name = n;
		age = a;
		gender = g;
		specialist = s;
```

```
	}

}

class Prog {
	public static void main(String args[]) {
		Employee e[] = new Employee[3];
		Specialist s[] = new Specialist[3];
		e[0] = new Employee("AAA", 12, "Male", 1);
		e[1] = new Employee("AAA", 12, "Male", 1);
		e[2] = new Employee("AAA", 12, "Male", 1);

		s[0] = new Specialist("AAA", 12, "Male", "AAA");
		s[1] = new Specialist("AAA", 12, "Male", "AAA");
		s[2] = new Specialist("AAA", 12, "Male", "AAA");
		System.out.println("Displaying the Employee Details");

		for(int i=0;i<3;i++)
		{

			e[i].display_person();
			e[i].display_emp();
		}

System.out.println("Displaying the Specialist Details");

		for(int i=0;i<3;i++)
		{

			s[i].display_person();
			s[i].display_emp();
			s[i].display_spec();
		}


	}
}
```

5) Write a frame program that inputs the three floating point numbers from the user and display the sum, avg, largest of three these numbers

```
import java.awt.*;
import java.awt.event.*;

import javax.swing.JApplet;

public class Prog extends JApplet implements ActionListener
{

   Label l1=new Label("First Number");
   Label l2=new Label("Second Number");
   Label l3=new Label("Third Number");
   Label l4=new Label("Result");
   TextField t1=new TextField(20);
   TextField t2=new TextField(20);
   TextField t3=new TextField(20);
   TextField t4=new TextField(20);
   Button b1=new Button("Sum");
   Button b2=new Button("Avg");
   Button b3=new Button("Max");
```

```
  Button b4=new Button("Cancel");

  public void init()
  {
       makeGUI();
  }
 public void makeGUI()
  {
              setLayout(new FlowLayout());
          add(l1);
     add(l2);
    add(l3);
   add(l4);

    add(t1);
    add(t2);
    add(t3);
   add(t4);

    add(b1);
   add(b2);
    add(b3);
    add(b4);

    b1.addActionListener(this);
    b2.addActionListener(this);
    b3.addActionListener(this);
    b4.addActionListener(this);


  }
 public void actionPerformed(ActionEvent e)
 {
            float n1=Float.parseFloat(t1.getText());
            float n2=Float.parseFloat(t2.getText());
            float n3=Float.parseFloat(t3.getText());

            if(e.getSource()==b1)
            {
                        t4.setText((n1+n2+n3)+"");
            }
            if(e.getSource()==b2)
            {
                        t4.setText((n1+n2+n3)/3)+"");
            }

            if(e.getSource()==b3)
            {
                        t4.setText((Math.max(Math.max(n1, n2),n3))+"");
            }

            if(e.getSource()==b4)
            {
                        System.exit(1);
            }

 }
}
```

6) Consider the string “NMAMIT” [Nitte] (Autonomous Institute) {Karkala}. Write synchronized thread program to print this string. Demonstrate the importance of synchronization

```
package part_a;
import java.io.*;

 class Callme {
        synchronized void call(String msg1,String msg2,String msg) {
                System.out.print(msg1+ msg);

                try {
                        Thread.sleep(1000);
                } catch(InterruptedException e) {
                        System.out.println("Interrupted");
                }
                System.out.print(msg2);
        }
}
class Caller implements Runnable {
        String msg,msg1,msg2;
        Callme target;
        Thread t;
        public Caller(Callme targ, String s1,String s2,String s) {
                target = targ;
                msg = s;
                msg1=s1;
                msg2=s2;
                t = new Thread(this);
                t.start();
        }
        public void run() {
                target.call(msg1,msg2,msg);
        }
}
class Q6 {
        public static void main(String args[]) {
                Callme target = new Callme();
                new Caller(target,"\"","\"" ,"NMAMIT");
                new Caller(target,"[","]" , "Nitte");
                new Caller(target, "(",")", "Autonomous Institute");
                new Caller(target, "{","}", "Karkala");
        }
}
```

7) Develop a set of methods, which work with an integer array. The methods to be implemented are:-
(i) min (which finds the minimum element in the array)
(ii) max (which finds the maximum element in the array – variable length argument)
(iii) scale (method to multiply the array – Use default arguments)
Place this in a package called p1. Let this package be present in a folder called “myPackages”, which is a folder in your present working directory. Write a main method to use the methods of package p1.

```
import mypack1.IntegerDemo;
import java.util.*;

class Prog
{
```

```
	public static void main(String args[])
	{

		System.out.println("Enter 10 elements");

		Scanner s = new Scanner(System.in);
		int a[] = new int [10];

		for(int i=0;i<10;i++)
		{
			a[i] = s.nextInt();
		}

		IntegerDemo obj = new IntegerDemo(a);

		obj.minimum();
		obj.maximum();

		System.out.println("Enter the scaling factor");
		int num = s.nextInt();

		obj.scaled(num);

		obj.printResult();


	}
}

package mypack1;

public class IntegerDemo {

	int arr[] = new int[10];
	int min , max ;
	int scale[] = new int[10];


	public IntegerDemo(int a[])
	{
		for(int i=0;i<10;i++)
		{
			arr[i]=a[i];
		}
	}

	public void minimum()
	{
		min  = 9999;

		for(int i=0;i<10;i++)
		{
			if(arr[i]<min)
			{
				min = arr[i];
			}
		}


	}
```

```
	public void maximum()
	{
		max  = -9999;

		for(int i=0;i<10;i++)
		{
			if(arr[i]>max)
			{
				max = arr[i];
			}
		}


	}


	public void scaled(int s)
	{
		for(int i=0;i<10;i++)
		{
			scale[i] = arr[i]*s;
		}
	}


	public void printResult()
	{
		System.out.println("Original aRRAY");

		for(int i=0;i<10;i++)
		{
			System.out.print(arr[i]+" ");
		}

		System.out.println("\n");

		System.out.println("Minimum is "+min);
		System.out.println("Maximum is "+max);
		System.out.println("Array after Scaling ");

		for(int i=0;i<10;i++)
		{
			System.out.print(scale[i]+" ");
		}
	}

}
```

8.Write a program to handle following exceptions: 1. Arithmetic 2.Array index out of Bound 3. Null pointer 4. Number Format 5. Illegal Access. Include one nested exception case.

```
public class Prog {
	  static void throwOne() throws IllegalAccessException {
	System.out.println("Inside throwOne.");
	throw new IllegalAccessException("demo");
	}
	  static void demoproc() {
		// create an exception
		  NullPointerException e = new NullPointerException("top layer");
```

```
                // add a cause
                e.initCause(new ArithmeticException("cause"));
                throw e;
        }
        public static void main(String args[]) {
                try {
                        int a =2;
                        int b = 42 / a;
                        System.out.println("a = " + a);
                        try {

if(a==2) {
int c[] = { 1 };

c[42] = 99;
}
} catch(ArrayIndexOutOfBoundsException e) {
System.err.println("Array index out-of-bounds: " + e);
}
} catch(ArithmeticException e) {
System.err.println("Divide by 0: " + e);
}

//Null pointer exception
try {
demoproc();
} catch(NullPointerException e) {
// display top level exception
System.err.println("Caught: " + e);
// display cause exception
System.err.println("Original cause: " + e.getCause());
}
//Number format exception
String str ="123!23";
                try{
                    float i = Float.valueOf(str);
                    //float i = Float.parseFloat(str);
                System.out.println("Value parsed :"+i);
                 }catch(NumberFormatException ex){
               System.err.println("Ilegal input");
                 }

//Illegal Access exception

                try {
                   throwOne();
                  } catch (IllegalAccessException e) {
                  System.err.println("Caught " + e);
                  }
}

}
```

9.Create a frame/Swing that is having a List View to select the city, Text box to enter the name, phone number. Radio button to choose the gender and check box to choose hobbies. Create an alert to display the selections made by the user.

```
import java.awt.*;
import java.awt.event.*;
```

```
import javax.swing.*;
import javax.swing.event.*;

public class prgm11 extends JApplet implements ActionListener
{
        String cities[] = {"Hong Kong","Mumbai","Chennai","Delhi"};

        String radio,name,phone_number,list_select;
        String check[] = new String[3];



        JList ls = new JList(cities);
    JTextField tx1 =   new JTextField(30);
    JTextField tx2 =   new JTextField(30);
    JRadioButton rb1 = new JRadioButton("Male");
    JRadioButton rb2 = new JRadioButton("Female");
    JCheckBox cb1 = new JCheckBox("Programming");
    JCheckBox cb2 = new JCheckBox("Debating");
    JCheckBox cb3 = new JCheckBox("Gyming");

    JLabel lb1 = new JLabel("Pick any city of your choice");
    JLabel lb2 = new JLabel("Enter Your Name");
    JLabel lb3 = new JLabel("Enter Phone Number");
    JLabel lb4 = new JLabel("Gender");
    JLabel lb5 = new JLabel("What are your Hobbies");

    JButton jb = new JButton("Done");

        public void init()
        {
                jb.addActionListener(this);
                makeGUI();

        }

        private void makeGUI()
        {
              //Create a list
                setLayout(new FlowLayout());
                add(lb1);
           add(ls);
           add(lb2);
                add(tx1);
                add(lb3);
                add(tx2);
                add(lb4);
                ButtonGroup bg1 = new ButtonGroup();
                bg1.add(rb1);
                bg1.add(rb2);

                add(rb1);
                add(rb2);

                add(lb5);

                add(cb1);
                add(cb2);
                add(cb3);
        }
```

```
	@Override
	public void valueChanged(ListSelectionEvent e) {
		// TODO Auto-generated method stub



	}

	@Override
	public void itemStateChanged(ItemEvent e) {
		// TODO Auto-generated method stub


	}

	@Override
	public void actionPerformed(ActionEvent e) {
		// TODO Auto-generated method stub

		if(rb1.isSelected())
		{
			radio = "MALE";
		}
		else
		{
			radio = "FEMALE";
		}

		name = tx1.getText();
		phone_number = tx2.getText();

		 if(cb1.isSelected())
		 {
			 check[0] = "Programming";

		 }
		  if(cb2.isSelected())
		 {
			 check[1] = "Debating";
		 }
		  if(cb3.isSelected())
		  {
			  check[2] = "Gyming";
		  }
		  int i = ls.getSelectedIndex();
		 list_select = cities[i];

		String temp = "Name = "+name+"\n Phone Number = "+phone_number+"\n";
		temp = temp+"Gender = "+radio+"\n";
		temp = temp+"Cities " list_select;

		JOptionPane.showMessageDialog(null, temp);


	}
}
```

10) Write a program to copy the contents of two files in to one file.

```
import java.io.*;
import java.util.*;

class prgm12
{
        public static void main(String args[])
        {
                try
                {
                 FileInputStream f1 = new FileInputStream("file1.txt");
                 FileInputStream f2 = new FileInputStream("file2.txt");
                 FileOutputStream f3 = new FileOutputStream("file3.txt");


                 int i;
       while((i=f1.read())!=-1)
       {
          f3.write(i);
       }


       f1.close();

       while((i=f2.read())!=-1)
       {
          f3.write(i);
       }

       f2.close();
       f3.close();




                }
                catch(Exception e)
                {
                        System.out.println("Exception Occured "+e);
                }
        }
}
```

11) Write a frame/swing program to display the contents of a file in a TextArea.

```
package part_a;

import java.awt.FlowLayout;
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;
import javax.swing.*;

public class Q11 extends JFrame {
        Q11() {
                setTitle("Read from File");
                setSize(500, 500);
                setLayout(new FlowLayout());
```

```
		JTextArea ta = new JTextArea(10, 20);
		add(ta);
		String temp = "";
		try {
			Scanner input = new Scanner(new File("src/sample.txt"));
			while(input.hasNext()) {
				temp += input.nextLine() + "\n";
			}
			ta.setText(temp);
			input.close();
		} catch (FileNotFoundException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		}
		setVisible(true);
	}
	public static void main(String args[]) {
		new Q11();
	}
}
```

12) Write a frame/swing program to write the contents of TextArea to a file.

```
package part_a;

import java.awt.FlowLayout;
import java.awt.event.*;
import java.io.*;
import javax.swing.*;

public class Q12 extends JFrame {
	JTextArea ta;
	Q12() {
		setTitle("Write to File");
		setSize(500, 500);
		setLayout(new FlowLayout());
		ta = new JTextArea(10, 20);
		add(ta);
		JButton bWrite = new JButton("Write");
		add(bWrite);
		bWrite.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent arg0) {
				try {
					PrintWriter output = new PrintWriter(new File("src/q12.txt"));
					output.println(ta.getText());
					output.close();
				} catch (FileNotFoundException e) {
					e.printStackTrace();
				}
			}
		});
		setVisible(true);
	}
	public static void main(String args[]) {
		new Q12();
	}
}
```

13) Write a frame/Swing program that is having and radio buttons to draw circle, line, rectangle.

```
package part_a;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class Q13 extends JFrame {
	Q13() {
		Shape s = new Shape();
		add(s);
		setTitle("Shapes");
		setSize(500,500);
		setVisible(true);
	}
	public static void main(String args[]) {
		new Q13();
	}
}

class Shape extends JPanel implements ActionListener {
	String str = "";
	Shape() {
		JRadioButton rbCircle = new JRadioButton("Circle");
		JRadioButton rbLine = new JRadioButton("Line");
		JRadioButton rbRectangle = new JRadioButton("Rectangle");
		setLayout(new FlowLayout());
		rbCircle.addActionListener(this);
		rbLine.addActionListener(this);
		rbRectangle.addActionListener(this);
		ButtonGroup grp = new ButtonGroup();
		grp.add(rbCircle);
		grp.add(rbLine);
		grp.add(rbRectangle);
		add(rbCircle);
		add(rbLine);
		add(rbRectangle);
	}

	public void actionPerformed(ActionEvent e) {
		str = e.getActionCommand();
		repaint();
	}

	public void paintComponent(Graphics g)
	{
		super.paintComponent(g);
		switch(str) {
		case "Circle": g.drawOval(200, 100, 50, 100); break;
		case "Line": g.drawLine(200, 200, 300, 300); break;
		case "Rectangle": g.drawRect(300, 200, 50, 50); break;
		}
	}
}
```

14) Another set of radio buttons to choose the background color. The color of frame background should be set to the selected color.

.

```
package part_a;
```

```
import java.awt.Color;
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.*;

public class Q14 extends JFrame implements ActionListener {
	Q14() {
		setTitle("Colors");
		setSize(500,500);
		JRadioButton rbRed = new JRadioButton("Red");
		JRadioButton rbBlue = new JRadioButton("Blue");
		JRadioButton rbGreen = new JRadioButton("Green");
		setLayout(new FlowLayout());
		rbRed.addActionListener(this);
		rbBlue.addActionListener(this);
		rbGreen.addActionListener(this);
		ButtonGroup grp = new ButtonGroup();
		grp.add(rbRed);
		grp.add(rbBlue);
		grp.add(rbGreen);
		add(rbRed);
		add(rbBlue);
		add(rbGreen);
		setVisible(true);
	}

	public void actionPerformed(ActionEvent arg0) {
		switch(arg0.getActionCommand()) {
		case "Red":
			getContentPane().setBackground(Color.red);
			break;
		case "Blue":
			getContentPane().setBackground(Color.blue);
			break;
		case "Green":
			getContentPane().setBackground(Color.green);
		}
		setVisible(true);
	}

	public static void main(String[] args) {
		new Q14();
	}
}
```

15) JDBC

```
import java.sql.*;
import java.io.*;
import java.util.*;

public class Jdbc1
{
	int ch;
	String usn,fname,lname;
	Scanner input=new Scanner(System.in);
	Statement stmt;
```

```
	Connection con;
	Jdbc1() throws Exception
	{
		Class.forName("com.mysql.jdbc.Driver");

	con=DriverManager.getConnection("jdbc:mysql://172.16.2.3/student","student","stude
nt");
		stmt=con.createStatement();

		while(true)
		{
		System.out.println("Enter 1.Insert 2.Delete 3.Upadate 4.DisplayAll 5.Display
6.Exit");
		ch=input.nextInt();
		switch(ch)
		{
			case 1:insert();break;
			case 2:delete();break;
			case 3:update();break;
			case 4:display("select * from sv");break;
			case 5:display(null);break;
			case 6:System.exit(0);
			default:System.out.println("Invalid Choice");
		}}

	}



	void insert() throws Exception
	{

		System.out.println("Enter USN :");
		usn=input.next();
		System.out.println("Enter Student First Name :");
		fname=input.next();
		System.out.println("Enter Student Last Name :");
		lname=input.next();
		stmt.executeUpdate("insert into sv
values('"+usn+"','"+fname+"','"+lname+"');");
		System.out.println("Record inserted sucessfully");
	}

	void delete() throws Exception
	{

		System.out.println("Enter USN to delete record :");
		usn=input.next();
		stmt.executeUpdate("delete from sv where usn='"+usn+"';");
		System.out.println("Record deleted sucessfully");
	}

	void update() throws Exception
	{

		System.out.println("Enter USN to update record:");
		usn=input.next();
		System.out.println("Enter Student First Name :");
		fname=input.next();
		System.out.println("Enter Student Last Name :");
		lname=input.next();
```

```
                stmt.executeUpdate("update sv set fname='"+fname+"',lname='"+lname+"'
where usn='"+usn+"';");
                System.out.println("Record updated sucessfully");
        }

        void display(String qry) throws Exception
        {
                ResultSet rs;
                if(qry==null)
                {
                System.out.println("Enter USN to display record :");
                usn=input.next();
                qry="select * from sv where usn='"+usn+"';";
                }
                rs=stmt.executeQuery(qry);
                System.out.println("");
                System.out.println("USN\tFirstName      LastName");
                while(rs.next())
                {

        System.out.println(rs.getString(1)+"\t"+rs.getString(2)+"\t\t"+rs.getString(3));
                }
                System.out.println("");
        }


        public static void main(String args[]) throws Exception
        {
                Jdbc1 obj=new Jdbc1();

        }
        }
```

16) TCP

Client

```
import java.io.*;
import java.net.*;
import java.util.*;
public class Tcp
{

public static void main(String[] args) throws Exception
{
  final int PORT = 1267;
  InetAddress host = InetAddress.getByName("127.0.0.1");
  Socket link = new Socket(host,PORT); //Step 1.

  Scanner input = new Scanner(link.getInputStream()); //Step 2.
  PrintWriter output = new PrintWriter(link.getOutputStream(),true); //Step 2.

  String  response;
 Scanner message = new Scanner(System.in);
 String a =message.next();
  output.println(a); //Step 3.
  response = input.nextLine(); //Step 3.
  System.out.println("\nSERVER> "+response);
  link.close(); //Step 4.
}

}
```

Server

```
import java.io.*;
import java.net.*;
import java.util.*;
public class Tcps
{
public static void main(String[] args) throws Exception
{
   final int PORT = 1266;
   System.out.println("Opening port...\n");
   ServerSocket servSock = new ServerSocket(PORT); //Step 1.
   Socket link = null;
   do
   {
      link = servSock.accept(); //Step 2.
      Scanner input =new Scanner(link.getInputStream());//Step 3.
      PrintWriter output =new PrintWriter(link.getOutputStream(),true); //Step 3.
      String message = input.nextLine(); //Step 4.
      output.println("tcp server: "  + message);

   }while (true);
   //link.close();
}


}
```

UDP

Server

```
import java.io.*;
import java.net.*;
public class Udps
{

public static void main(String[] args) throws Exception
{
   final int PORT = 1263;
   DatagramSocket datagramSocket;
   DatagramPacket inPacket, outPacket;
   byte[] buffer;

   System.out.println("Opening port...\n");
   datagramSocket = new DatagramSocket(PORT); //Step 1.

   do
   {
   buffer = new byte[256]; //Step 2.
   inPacket = new DatagramPacket(buffer, buffer.length); //Step 3.
   datagramSocket.receive(inPacket);//Step 4.

   InetAddress clientAddress = inPacket.getAddress(); //Step 5.
   int clientPort = inPacket.getPort(); //Step 5.

   String messageIn = new String(inPacket.getData(),0,inPacket.getLength()); //Step 6.
   System.out.println("Message received.");

   String messageOut = "Message: " + messageIn;
   outPacket = new DatagramPacket(messageOut.getBytes(),
         messageOut.length(),clientAddress,clientPort); //Step 7.
   datagramSocket.send(outPacket); //Step 8.
```

```
   }while (true);
}
}
```

Client

```
import java.io.*;
import java.net.*;
import java.util.*;
public class Udpc
{
public static void main(String[] args) throws Exception
{
   InetAddress host;
   final int PORT = 1263;
   DatagramSocket datagramSocket;
   DatagramPacket inPacket, outPacket;
   byte[] buffer;
   host = InetAddress.getByName("127.0.0.1");

   //Step 1...
   datagramSocket = new DatagramSocket();
   String message="Test Msg";
   outPacket = new DatagramPacket(message.getBytes(),message.length(),
   host,PORT); //Step 2.
   //Step 3...
   datagramSocket.send(outPacket);
   buffer = new byte[256]; //Step 4.
   inPacket = new DatagramPacket(buffer, buffer.length);//Step 5.
   //Step 6...
   datagramSocket.receive(inPacket);
   String response = new String(inPacket.getData(),0, inPacket.getLength()); //Step 7.
   System.out.println("\nSERVER> "+response);
   System.out.println("\n* Closing connection... *");
   datagramSocket.close(); //Step 8.
}
}
```